融合语言特征的抽象式中文摘要模型 *

胡德敏，王荣荣

(上海理工大学 光电信息与计算机工程学院，上海 200093)

摘　要：为了解决传统抽象式摘要模型生成的中文摘要难以保存原文本语义信息的问题，提出了一种融合语言特征的抽象式中文摘要模型。模型中添加了拼接层，将词性、命名实体、词汇位置、TF-IDF 等特征拼接到词向量上，使输入模型的词向量包含更多的维度的语义信息来确定关键实体。结合指针机制有选择地复制原文中的关键词到摘要中，从而提高生成的摘要的语义相关性。使用 LCSTS 新闻数据集进行实验，取得了高于基线模型的 ROUGE 得分。分析表明本模型能够生成语义相关度较高的中文摘要。

关键词：抽象式摘要模型；语言特征；关键实体；词向量

中图分类号：TP391.1　　doi: 10.19734/j.issn.1001-3695.2018.07.0531

Abstractive Chinese summarization model with linguistic features

Hu Demin, Wang Rongrong

(*School of Optical-Electrical & Computer Engineering, University of Shanghai for Science & Technology, Shanghai* 200093, *China*)

Abstract: In order to solve the problem that the Chinese summarization generated by traditional abstractive models can hardly preserve the semantic information of the original text, this paper proposed an abstractive Chinese summarization model with linguistic features. A connection layer is added to the model, and features such as part of speech, named entity, word position, and TF-IDF are spliced into the word vector, so that the word vector of the input model contains more semantic information to determine the key entity. The pointer mechanism allows model selectively copy the keywords in source text into the summarization to improve the semantic relevance between source text and summarization. Evaluates this model on LCSTS dataset, and obtains a higher ROUGE score than the baseline model. The analysis shows that the model can generate Chinese summarization with higher semantic relevance.

Key words: abstractive summarization model; linguistic features; key entities; word vector

0　引言

生成简洁凝练，语义连贯，保留关键信息的总结是自动文本摘要的最终目标。根据对信息的抽取方式的不同，可将文本自动摘要技术主要分为两大类：抽取式文本摘要生成方式和抽象式文本摘要生成方式[1]。目前的中文摘要研究大多使用抽取式方法，根据语言特征计算句子权重，复制比较重要的句子组成摘要，但这种方法没有考虑句子间的连贯性，不能完整的表达文章的含义；抽象式文本摘要生成方法应用神经网络模型，通过对大量的数据进行训练，生成深入理解原文的新句子。

与抽取式方法提取原文的句子作为摘要不同的是，抽象式摘要方法不是简单地从原文中提取的一些现有的段落或句子，而是对文档的主要内容进行了压缩解释，重新措辞，使用了原文档中未现的词汇来生成摘要。抽象式方法生成的摘要更接近于人工生成的摘要。Sutskever 等人[2]提出的 sequence-to-sequence 模型（简称 seq2seq）和 Bahdanau 等人[3]提出的 Attention 机制，推动了抽象式自动摘要的发展。但抽象式摘要方法仍处于早期阶段，存在一定的局限性，比如，依赖大规模、高质量的训练集来训练模型；适用于短文本摘要生成，在长文本上的摘要效果较差；生成的摘要语义相关性较低，往往存在语法和语义错误。

为了提高抽象式摘要与原文本的相关度，本文提出了一种融合语言特征的抽象式摘要模型（简称 LF_model）。本文认为抓住原文中的关键实体可以使摘要更加贴近文章的主题，考虑了输入模型的词汇的语言特征对摘要质量的影响，将原文本的词性标注，命名实体，词汇位置，TF-IDF 等特征向量化后与原始词向量拼接在一起构建输入模型的词向量，使输入模型的向量有更多维度的含义来抓取原文中的关键实体。考虑到未登录词大多是原文本中的命名实体，解决 OOV（out of vocabulary）问题有助于模型输出原文中的关键实体，本模型结合 Gulcehre 等人[5]提出的 Pointer 机制选择性地复制原文的词汇到摘要中，从而生成与原文本语义相关度高的摘要。使用 LCSTS 新闻数据集来训练模型，并将生成的摘要的评价得分同基线模型进行了对比，取得了比基线模型表现更好的实验结果。

1　相关工作

当前采用抽取式方法生成摘要的技术相对比较成熟，中文摘要的研究大多采用抽取式的方法，根据句子的各种文本特征，例如句子长度、句子位置、句子与文章标题的相似度、语言规则等来计算句子权重，根据句子的总权重给句子排序，

收稿日期：2018-07-28；**修回日期：**2018-09-12　**基金项目：**国家自然科学基金资助项目（61170227，61472256）；上海市教委科研创新重点资助项目（12zz17）；上海市一流学科建设项目（S1201YLXK）

作者简介：胡德敏（1963-），男，上海人，副教授，博士，主要研究方向为计算机网络、分布式计算、云计算（deminhu@usst.edu.cn）；王荣荣（1994-），女，硕士，主要研究方向为自然语言处理、深度学习.

选取权重高的句子作为摘要句。

Rush 等人[6]第一次使用 Seq2Seq+Attention 模型进行句子摘要任务，其中 Seq2Seq 模型也称为 Encoder-Decoder 模型，使用一个循环神经网络作为编码器读取输入的句子，将整个句子的信息压缩到一个连续的中间语义向量中。再使用另一个循环神经网络作为解码器读取这个中间语义向量，将其解压为目标语言的一个句子[3]。Attention 机制，使模型在输出端的某个节点将注意力集中在输入部分的某一个特定部分，而不是如以往的工作将输入部分作为一个整体均等的送入每一个输出端[3]，便于理解输入序列中的信息是如何影响最后生成的序列的。且作者提出了利用 Gigaword 构建大量平行句对的方法，使得利用神经网络训练成为可能，但该模型更适用于为一个句子生成摘要。Lopyrev 等人[7]描述了一个使用 LSTM（Long Short-Term Memory）作为循环神经网络计算单元，联合注意力机制来生成新闻摘要的应用，但未处理 OOV（Out of vocabulary）问题。为了处理 OOV 问题，Gu 等人[8]提出了一种合并复制机制，允许一部分摘要复制原文中的内容。Nallapati 等人[9]研究了关键词对于自动文摘所起到的关键作用，使用了 Feature-rich Encoder 来尝试抓住句子的关键概念和关键实体。还提出了 Generator-Pointer 机制，使编码器能够生成原文中的句子。Romain 等人[1]提出了内部注意力机制和新的训练方法，有效的提升了文本摘要生成的质量。Hu 等人[10]构建了一个大规模的中文语料库，并提供了基线，为研究中文摘要提供了便利。Ma 等人在提高抽象式摘要的质量上做了很多尝试，在文献[4]中提出一种引入了相似性评估组件的模型来提高语义相关性，在文献[11]中，使用自动编码器作为辅助监控器，来改进中文新闻文本的文本表示。

本文借鉴抽取式方法，研究了语言特征对摘要的影响，提出了一种融合语言特征的抽象式中文摘要模型。使用引入注意力机制的 encoder-decoder 模型作为基础框架，在模型的输入端添加了拼接层，用于将原文词汇的词性、命名实体、词汇位置和 TF-IDF 等特征与原始词向量融合在一起，构成输入模型的词向量。使用 Bi-LSTM(bi-directional long short-term memory)为编码器从正反两个方向编码生成中间语义向量，单向 LSTM 为解码器读取中间语义向量生成目标序列，模型结合 pointer 机制，在每个解码步骤中使用开关函数来决定是正常预测词表还是复制原文中的词，最终生成与原文语义相关度高的摘要。

2　模型

2.1　基于语言特征的词汇表示

抽象式摘要方法，通过对大量的数据进行训练而预测生成的新的摘要句子，摘要句子中会出现在输入文档中未出现的句子。抽象式方法考虑了摘要句子的语法正确性和连贯性，而忽略了生成摘要与原文档的语义相关性，从而导致生成与原文无关的摘要[4]。如图 1 所示，RNN 生成的摘要语句通顺但与输入的原文没有太大的关联。为了解决这个问题，本文抽取词汇的词性、命名实体、词汇位置和 TF-IDF 等特征来抓住原文本的关键实体。本文认为将词性、命名实体等语言特征融入词向量，可以改进模型避免语法错误并生成良好的摘要。词汇的 TF-IDF 特征值能够评估该词汇对原文的重要程度。除此之外，根据新闻的特点，将词在新闻文本中的位置也作为一项特征融入到了词向量中。

1)词性　词性是词汇基本的语法属性，决定了词汇的语义倾向性[12]。提取词性特征有助于探究和识别相邻词之间的关系和化解自然语言中一词多义的问题，对语义理解具有重要的作用。研究发现，在摘要任务中名词和动词相对于其他词性的词汇往往更能体现原文的关键信息。因此本文对训练集中的词汇进行词性标注，将词性进行 Embedding 表示后与词向量拼接，使词汇的词向量包含词性特征。

原文：昨晚，中联航空成都飞北京一架航班被发现有多人吸烟。后因天气原因，飞机备降太原机场。几名乘客在舱门边吸烟被发现。有乘客要求重新安检，机长决定继续飞行，引起机组人员与未吸烟乘客冲突。目前中联航空正联系机组进行核实。 人工摘要：成都飞北京航班多人吸烟机组人员与未吸烟乘客冲突。 RNN：中联航空机场发生爆炸致多人死亡。

图 1　一个用 RNN 生成的低语义相关的摘要实例。

Fig. 1　Example of RNN generated summary with low relevance

2)命名实体　命名实体就是人名、地名、机构名、专有名词等具有特定意义的实体[12]。在摘要任务中，命名实体是文本信息的主要载体，识别出文章中的命名实体不仅有助于模型确定代表文章主题的关键实体还能帮助模型处理 OOV 问题，因此，本文对语料库进行命名实体识别，将命名实体 Embedding 后与词向量拼接，使词汇的词向量拥有命名实体特征。

3)词汇位置　词汇在文本中所处的位置是新闻文本的另一个一个重要特征，新闻类文章的第一句或第一段往往会覆盖整片文章的主旨信息，距离文章开始位置越近的词汇越接近文章的主题，因此计算词汇的位置特征如式(1)所示来提高摘要的质量。

$$Loc_i = (1 + n - l_i) / n \tag{1}$$

其中:Loc 代表词汇的位置特征，l_i 代表新闻文本中第 i 个词汇的位置，n 代表该新闻文本中总的词汇数目，Loc 的值越大，证明该位置的词汇越重要，。

4)TF-IDF　词频-逆文档频率(term frequency-inverse document frequency，简称 TF-IDF)是一种统计方法，用以评估一个特定词语对于一个语料库的其中一份文本的重要程度。TF 为词频，用来统计词汇在该文本中出现的频率。IDF 为逆文档频率，用于识别某一词汇在整个语料库中的重要性。TF-IDF 为词频与逆文档频率的乘积，TF-IDF 越大，则说明这个词对这篇文章的区分度就越高。TF-IDF 的计算公式如下所示：

$$tf-idf_{i,j} = tf_{i,j} * idf_i \tag{2}$$

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{3}$$

$$idf_i = \log\left(\frac{N}{DF_i + 1}\right) \tag{4}$$

在式(3)中，$n_{i,j}$ 为词汇 t_i 在文章 d_j 中出现的次数，$\sum_k n_{k,j}$ 为文章 d_j 中所有的词汇数目。在式(4)中，N 为语料库的文档总数，DF_i 为语料库中包含词汇 t_i 的文档数目。当词汇未出现在语料库中时 DF_i 为零，为了避免分母为零，将 DF_i+1 作为分母来计算 IDF。

本文将词 embedding 成原始词向量，在原始词向量后添加经过 embedding 后的 POS、NER 和 Los、TF-IDF 等特征。

于是输入编码器的词汇被形象的表示为

$$x_i=\{r_i^w,r_i^{pos},r_i^{ner},r_i^{loc},r_i^{tf-idf}\} \tag{5}$$

其中: r^w 代表词汇的原始词向量，r^{pos} 代表词的词性标注的 embedding 向量，r^{ner} 代表词的命名实体识别的 embedding 向量，r^{loc} 代表词的位置特征，r^{tf-idf} 是词的 TF-IDF 特征。拼接层将这五种向量拼接起来作为最终输入编码器的向量。

2.2 LSTM 循环神经网络

基于神经网络的 seq2seq 模型由两部分组成，编码器和解码器 ，LSTM 长短期记忆网络是一种特殊的循环体结构，LSTM 计算单元添加了一种门机制来解决标准 RNN 模型的梯度消失问题。LSTM 的计算单元的结构如图 2 所示。在很多任务上，采用 LSTM 结构的循环神经网络比标准的循环神经网络表现更好。本模型使用 LSTM 作为编码器和解码器，LSTM 在 t 时刻的隐藏层状态 h_t 的计算公式如下所示：

$$\begin{bmatrix} i_t \\ f_t \\ O_t \\ c_t \end{bmatrix}=\begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \tan h \end{bmatrix} W\cdot[h_{t-1},x_t] \tag{6}$$

$$C_t=f_t\odot C_{t-1}+i_t\odot c_t \tag{7}$$

$$h_t=O_t\odot\tan h(C_t) \tag{8}$$

其中：i_t 指输入门，f_t 指忘记门，O_t 指输出门，c_t 更新候选向量，W 代表被学习的权值矩阵，σ 代表激励函数，$\odot$ 代表逐点运算操作。

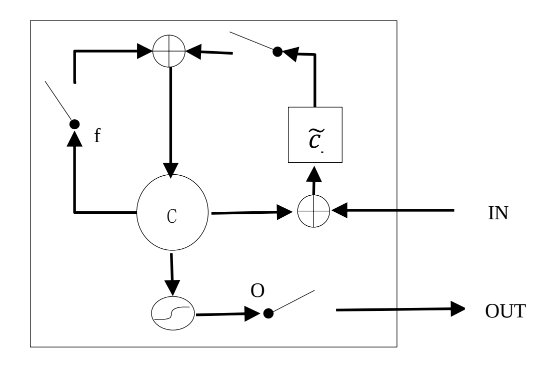


图 2　LSTM 计算单元结构图

Fig. 2　Illustration of LSTM

2.3 融合语言特征的神经网络模型

融合语言特征的神经网络模型如图 3 所示。本模型在输入层添加了一个拼接层，用于将词汇的原始词向量与词性、命名实体、词汇位置、TF-IDF 等语言特征拼接起来生成最终输入模型的词向量。原始词向量进入拼接层，拼接层根据式（1）计算该文章中词汇的位置信息，根据式（2）计算该文章中词汇的 TF-IDF 特征值，将每个词汇的词性标记和命名实体标记映射为 POS embedding 和 NER embedding。将每个词汇的 POS embedding、NER embedding、Loc、IF-IDF 与原始词向量拼接在一起，最终构成一个 512 维的向量 $x_i=\{r_i^w,r_i^{pos},r_i^{ner},r_i^{loc},r_i^{tf-idf}\}$。

编码器将整个原文本压缩成一个连续的向量，学习原文本的每个单词的矢量表示。本文使用 Bi-LSTM 作为编码器，向前 LSTM 从左向右读取输入序列 $x=(x_1,\dots,x_m)$，生成隐藏状态序列 $(\overrightarrow{h_1},\dots,\overrightarrow{h_m})$。向后的 LSTM 反向读取输入序列，生成 $(\overleftarrow{h_1},\dots,\overleftarrow{h_m})$。在每个时间步骤中连接向前和向后的 LSTM 的隐藏状态得到 $(h_1,\dots,h_m)$，其中 $(h_t=[\overrightarrow{h_t};\overleftarrow{h_t}])$，隐藏状态 h_t 包含了词向量正反两个方向的信息。

本文使用单向 LSTM 作为解码器，$h_i^{'}$ 表示解码器在 i 时刻的隐藏状态，如式（9）所示。在每个解码步骤中，引入了注意力机制使注意力集中在输入序列的某一个特定部分，内容向量 c_i 用于对系统所关注的词进行编码以生成下一个摘要词，如式（10）～（12）所示。

$$h_i^{'}=f(h_{i-1}^{'},y_{i-1},c_i) \tag{9}$$

$$e_{ij}=a(h_{i-1}^{'},h_j) \tag{10}$$

$$a_{ij}=\frac{exp(e_{ij})}{\sum_{k=1}^{m}exp(e_{ik})} \tag{11}$$

$$c_i=\sum_{j=1}^{m}a_{ij}h_j \tag{12}$$

其中：e_{ij} 是输入隐藏状态 h_j 和输出隐藏状态 $h_{i-1}^{'}$ 的注意力得分，a_{ij} 为标准化后的注意力得分。

模型结合 Pointer 机制，在解码端使用了一个开关(switch)函数，决定在每个解码步骤是正常的预测词表生成摘要词 y_i^g，还是复制原文的词 y_i^p 作为摘要词。如果 $u_i=1$，表示开关打开，正常预测词表。如果 $u_i=0$，表示开关关闭，指向原文的一个位置，将指针指向的词作为输出。开关打开的概率的计算公式如下：

$$p(u_i=1)=\sigma\left(v^s\cdot\left(W_h^s h_i^{'}+W_e^s E[o_{i-1}]+W_c^s c_i+b^s\right)\right) \tag{13}$$

其中：$h_i^{'}$ 是隐藏状态，$E[o_{i-1}]$ 上一个时间步的词向量，c_i 是上下文的权重，W_h^s，W_e^s，W_c^s，b^s 和 v^s 是开关的参数。使用文档中单词的注意力分布作为采样指针的分布。

$$p_i^a(j)\propto exp\left(v^a\cdot\left(W_h^a h_{i-1}^{'}+W_e^a E[o_{i-1}]+W_c^a h_j+b^a\right)\right),$$

$$p_i=argmax\left(p_i^a(j)\right)\ for\ j\in\{1,\dots,m\} \tag{14}$$

$p_i^a(j)$ 是解码的第 i 个时间步指向原文位置 j 的概率，h_j 是编码器在位置 j 处的隐藏状态，p_i 是摘要位置 i 处的指针值。联合式（13）（14）得到最终输出 y_i 的概率为

$$p(y_i)=p(u_i=1)\,p(y_i|u_i=1)+p(u_i=0)\,p(y_i|u_i=0) \tag{15}$$

在训练时过程中，当摘要中出现未登录词时，为模型提供显式的指针信息，当生成摘要的第 i 个位置的单词是未登录词或命名实体时 u_i 被设置为 0。优化似然函数如下所示：

$$\log p(y|x)=\sum_i\left(log\{p(y_i|y_{-i},x)\,p(u_i)\}+\log p(p(i)|y_{-i},x)(1-p(u_i))\right) \tag{16}$$

在测试时，模型在每个时间步，根据估计的开关函数的概率 $p(u_i)$ 来决定是正常的预测词表还是指向原文中的一个位置。

2.4 摘要生成流程

新闻摘要的生成流程如图 4 所示。

a）读取新闻文本 text。

b）预处理：为新闻文本分词，分词后生成词汇表 Vocab，为 Vocab 中的词生成所对应的词性标志和命名实体标志。

c）计算输入序列长度 m=count(Vocab)，创建输入模型的向量数组 new Text_Matrix[m][]，将 Vocab 中的词汇、词性标

志和命名实体标志向量化，获得原始词向量 r^w 、词性标志向量 r^{pos} 和命名实体标志的向量 r^{ner} ，根据式(1)计算词的位置特征值 r^{loc} 、根据式(2)计算每个词的 TF-IDF 值 r^{tf-idf} 。

d）拼接语言特征向量 concatenate($r^w, r^{pos}, r^{ner}, r^{loc}, r^{tf-idf}$) → Text_Matrix[i][]。

e）计算编码层隐藏状态 h_t ，根据式(16)计算输出 y_t 的概率

使用 Beam Search 算法 select top 5 score 迭代预测摘要。

f）输出新闻摘要。

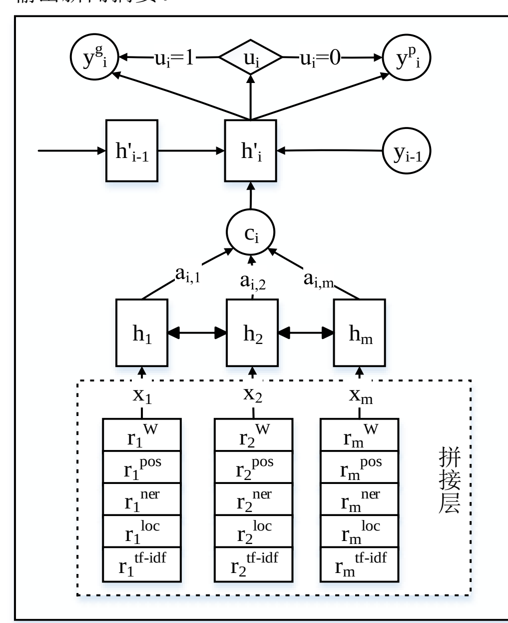


图 3　融合语言特征的神经网络模型

Fig. 3　Abstractive model with linguistic features

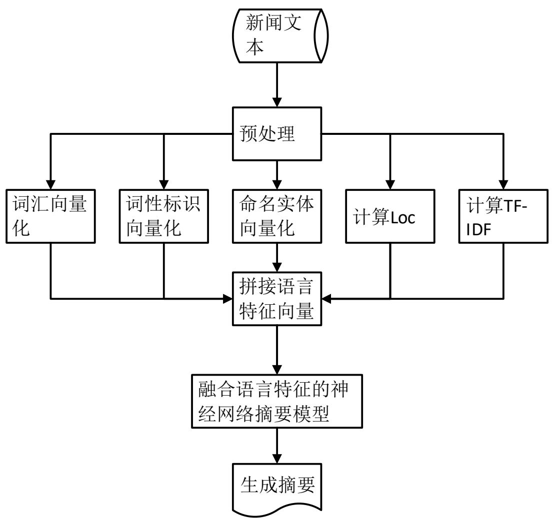


图 4　摘要生成流程图

Fig. 4　Summarization generating process

3　实验

3.1　数据集

数据集的质量、内容和规模都直接影响摘要的生成效果，LCSTS 是当前最大规模的中文数据集，是从新浪微博上爬取过滤得到的，包含了超过 240 多万对的新闻文本以及摘要，该数据集质量高，涵盖领域广。数据集来源于具有较大影响力的官方微博，例如，“人民日报”“经济观察报”“国防部”等[10]，这些新闻内容书写规范，语句通顺，几乎不存在错别字，非常适合深度学习模型的研究。LCSTS 的训练集有 240 万多对，验证集有 1 万多对，测试集有 1 千多对，而且验证集和测试集用人工标注了正文和标题之间的相关度，并且从 1-5 打分，分数越高越好。本文采用 LCSTS 中给出的数据集来训练模型，使用测试集中 3 分以上的数据来测试模型。

3.2　预处理

本文首先对语料库中的文本进行预处理。词是最小的能够独立运用的语言单位，本文使用基于分词的词向量进行实验，使用 Stanford CoreNLP 工具包对语料进行分词、词性标注和命名实体识别。表 1 给出了一篇新闻内容经过预处理的示例。统计数据集内每个词汇的出现频率，按照词频的顺序对单词进行排序，从中选取使用频率高的词来构建词汇表，词汇表大小为 50000。词汇表中的<unk>符号被用来代替测试数据集中未出现在训练词汇表中的词。使用 Gensim 工具包，对包含动词、形容词、名词、代词等 40 种中文词性标记进行 embedding 处理生成 POS embedding，对包含有人名、机构名、地名、时间、日期、货币、百分比和非命名实体等 8 种类型命名实体标记进行 embedding 处理生成 NER embedding，使用 CBOW 模型来生成每个词所对应的原始词向量。将每个词的原始词向量与所对应的 POS embedding 和 NER embedding 一一对应。

表 1　文本预处理示例

Table 1　Example of pre-process

新闻内容	总理 18 日在美药典公司餐厅与 10 家进驻自贸区的中外企业家座谈，请他们给自贸区各项改革“打分”。他对 10 位参会企业家说：“希望我们在留有饭菜余香中进行的座谈会，不仅 friendly(友好)，而且 frankly(坦率)，有什么问题直来直去讲出来。
分词结果	总理 18 日 在 美 药典 公司 餐厅 与 10 家 进驻 自贸区 的 中外 企业家 座谈 请 他们 给 自贸区 各项 改革 打分 他 对 10 位 参会 企业家 说 希望 我们 在 留有 饭菜 余香 中 进行 的 座谈 会 不仅 friendly 友好 而且 frankly 坦率 有 什么 问题 直来直去 讲 出来
词性标注结果	nr n m q p a n n n p m n v nz uj j n v v r v nz r vn v r p un q v n v v r p v n n f v uj n c en ad c en an v r n l v v
命名实体识别结果	Person O O O O O O O O O O O O Loction O O O O O O O O Location O O O O O O O O O O O O O O O O O O O O O O O O O O O O O O O O O

3.3　评价指标

如何有效合理地评价文本摘要的生成效果是一个很难的问题，当前的文本摘要评价方法分为两类，一种是内部评价方法，将获得的摘要与参考摘要进行对比，根据两者的相似性进行评价。与参考摘要越吻合，说明摘要的质量就越高；另一种是外部评价方法，将摘要应用于特定的任务，根据摘要提高这项任务的效果来评价生成文摘的效果。本文使用了

目前流行的内部评价方法 ROUGE。ROUGE 评价方法是 Lin 等人[13]提出的一种自动文本摘要评价方法。其通过统计自动生成的摘要与参考摘要之间的重叠基本单元的数目来评价文摘的质量。本文使用的参考摘要为数据集中的人工摘要。ROUGE 常用的评价标准有 ROUGE-N 和 ROUGE-L。其中 ROUGE-N 表示系统生成摘要的 n-gram 召回率，ROUGR-L

表示系统摘要和参考摘要的最大公共序列。ROUGE-N 的计算方法如式（17）所示。

$$\text{ROUGE}-\text{N}=\frac{\sum_{S\in R}\sum_{gram_n\in S}Count_{match}(gram_n)}{\sum_{S\in R}\sum_{gram_n\in S}Count(gram_n)} \tag{17}$$

其中：$Count_{match}(gram_n)$ 表示系统输出的摘要与参考摘要重叠的 n-gram 的个数，R 表示参考摘要。ROUGE-L 考虑了摘要中词汇的顺序，评价更合理。本文使用了 Lin 提供的标准工具包，选用了 ROUGE-1，ROUGE-2，和 ROUGE-L 来评价本模型生成的摘要质量。

3.4　实验设置

本文使用 TensorFlow 框架进行实验，原始词向量的维度是 350，经过拼接层后输入编码器的词向量的维度是 512，隐藏层的大小为 512，批次大小为 64。使用 Adam 优化器，默认设置为 $\alpha=0.001$，$\beta_1=0.9$，$\beta_2=0.999$，$\epsilon=1\times10^{-8}$。在测试时，为了得到最符合语言模型的摘要，本文选择使用集束搜索(Beam Search)算法，集束大小设置为 5 来生成摘要。

3.5　实验结果及分析

本文使用 LCSTS 中文数据集来验证模型的生成效果，将实验结果分别与 Hu 等人[10]提出的 RNN context 模型、Gu 等人[8]提出的 COPYNET 模型和 Ma 等人[11]提出的 SRB 模型的实验结果进行对比。取得了比上述模型更高的 ROUGE 得分，如表 2 所示。

表 2　在 lcstc 数据集上的 ROUGE 得分表

Table 2　ROUGE score on LCSTS dataset

model	R-1	R-2	R-L
RNN context(W)	26.8	16.1	24.1
RNN context(C)	29.9	17.4	27.2
COPYNET(W)	35.0	22.3	32.0
COPYNET(C)	34.4	21.6	31.3
SRB(C)	33.3	20.0	30.1
LF_model	36.2	23.6	32.9

RNN context 模型是 Hu 等人[10]使用的引入上下文的摘要生成模型。作者对 LCSTS 数据集分别进行词语级别分词处理和字符级别分词处理后进行对比，实验结果显示使用

字符级别分词处理优于基于词语级别分词处理结果。这是因为，根据字符分词生成的词典远远小于根据词语分词生成的词典。字符词典能覆盖更多的原文内容，有效减少了 OOV 问题。RNN context 模型的结果成为后来使用 LCSTS 数据进行中文摘要研究的基线。

COPYNET 模型是 Gu 等人[8]提出的一种能够解决 OOV 问题的模型。COPYNET 在 Seq2seq+Attention 模型的基础上引入了拷贝机制，允许部分摘要复制原文中的内容。在基于分词表示的摘要任务上取得了更高的 ROUGE 得分。

SRB 模型是 Ma 等人[11]提出了一种基于语义相关性的神经模型，用来鼓励文本和摘要之间的语义相似性。SRB 在基于字符表示的摘要任务中能够生成与原文语义相关度较高的摘要，但未考虑 OOV 问题获得了低于 COPYNET 模型的 ROUGE 得分。

由于汉字通常有多重语义，使用基于字符的向量可能会误解句子的意义。本文认为基于词语的表示可以更准确的捕捉文章的语义，因此本模型使用基于分词表示的词向量输入模型。除此之外，本模型结合 Pointer 机制选择性地输出原文中的词汇，不仅能够有效地解决 OOV 问题还能输出原文中的关键词。

本文使用融合了词性、命名实体、IF-IDF 值和 Loc 等语言特征的 LF_model 生成的摘要与未融合语言特征的 lack feature model 生成的摘要进行了对比，结果如表 3 所示。

实验结果表明，使用融合语言特征的模型获得了更高的 ROUGE 得分。证明了语言特征对生成摘要的质量的影响，融入词性、命名实体、位置特征、TF-IDF 等语言特征扩展了词向量的维度，使输入模型的词向量包含了更多的语义信息。词性信息使模型关注原文中动名词，识别出命名实体有助于模型输出 OOV 词，IF-IDF 帮助模型识别语料中的重要词汇，位置信息使模型关注词汇在文章中的位置对摘要的影响。最终生成更贴近原文主题的摘要。

表 3　融合语言特征的 rouge 得分表

Table 3　ROUGE score with linguistic features

model	R-1	R-2	R-L
lack feature model	35.4	21.1	30.5
LF_model	36.2	23.6	32.9

图 5 是使用本模型生成的摘要实例，通过观察可以发现传统的 RNN 模型生成的摘要可读性差，语义相关性较低，存在 OOV 问题。本模型生成的摘要抓住了该条新闻的关键词汇“　　总理”“企业家和自贸区”，可读性更强，并且缓解了 OOV 问题。结果表明本文提出的模型生成了更接近人工生成的摘要，生成的摘要与原文内容的语义相关度较高。

原文：李克强总理 18 日在美药典公司餐厅与 10 家进驻自贸区的中外企业家座谈，请他们给自贸区各项改革“打分”。他对 10 位参会企业家说：“希望我们在留有饭菜余香中进行的座谈会，不仅 friendly(友好)，而且 frankly(坦率)，有什么问题直来直去讲出来。”

人工摘要：李克强邀 10 企业给上海自贸区打分。

RNN context(W)：李克强在每 UNK 公司给自贸区打分：有什么问题 UNK 讲出来。

Our model：李克强总理与中外企业家座谈自贸区改革。

图 5　本模型生成的摘要实例

Fig. 5　Example of summary generated by our model

4　结束语

考虑到输入模型的向量对摘要的影响，本文提出了一种融合语言特征的神经网络摘要模型来解决语义相关度低的问题。模型中的拼接层，用于将语言特征融入词向量，使模型能够抓取原文本中的关键实体，pointer 机制用来解决 OOV 问题和输出关键实体。在 LCATS 数据集上的实验结果表明，本文提出的模型不仅生成高于基线模型的 ROUGE 得分，还能够抓住原文本的关键实体缓解 OOV 问题，生成与原文本语义相关度较高的摘要。

参考文献：

[1] Paulus R, Xiong Caiming, Socher R. A deep reinforced model for abstractive summarization. [EB/OL]. (2015). http://cn.arxiv.org/abs/1705.04304.

[2] Sutskever I, Vinyals O, Le Quoc V. Sequence to sequence learning with neural networks [EB/OL]. (2014-12-14). https://arxiv.org/pdf/1409.3215v3.pdf.

[3] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate [EB/OL]. (2014) . http://cn.arXiv.org/abs/1409. 0473.

[4] Ma Shuming, Sun Xu, Xu Jinging. Improving semantic relevance for sequence-to-sequence learning of Chinese social media text summarization [C]//Proc of Meeting of Association for Computational

Linguistics. 2017: 635-640.

[5] Gulcehre C, Ahn S, Nallapati R. Pointing the unknown words [EB/OL]. (2016) . http//cn.arxiv.org/abs/1603. 08148.

[6] Rush A M, Chopra S, Weston J. A neural model for abstractive sentence summarization [C]//Empirical Methods in Natural Language Processing. 2015: 379-389.

[7] Lopyrev K. Generating new headlines with recurrent neural networks [J]. Computer Science, 2015.

[8] Gu Jiatao, Lu Zhengdong, Li Huang. Incorporating copying mechanism in sequence-to-sequence learning [C]// Proc of the 54 th Annual Meeting of the Association for Computational Linguistics. ACL. 2016: 1631-1640.

[9] Nallapati R, Zhou Bowen, dos Santos C. Abstractive text summarization using sequence-to-sequence RNNs and beyond [C]// Proc of the 20th SIGNLL Conference on Computational Natural Language Learning. 2016: 280-290.

[10] Hu Baotian, Chen Qingcai, Zhu Fangze. LCSTS: a large scale Chinese short text summarization dataset [J]. Computer Science, 2015, 2667-2671.

[11] Ma Shuming, Sun Xu, Lin Junyang. Autoencoder as assistant supervisor: improving text representation for chinese social media text summarization [EB/OL]. (2018) . http//cn. arxiv. org/abs/1805. 04869.

[12] 宗成庆.统计自然语言处理 [M].2 版. 北京: 清华大学出版社, 2013: 150-175. (Zong Chengqing, Statistical natural language processing [M]. 2nd ed. Beijing: Tsinghua University, 2013: 150-175.)

[13] Lin C Y, Hovy E. Automatic evaluation of summaries using *n*-gram co-occurrence statistics [C]//Proc of Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology. Association for Computational Linguistics, 2003: 71-78.

[14] Tan Jiwei, Wan Xiaojun, Xiao Jianguo. Abstractive document summarization with a graph-based attentional neural model [C]// Proc of Meeting of the Association for Computational Linguistics. 2017: 1171-1181.

[15] See A, Liu P J, Manning C D. Get to the point: summarization with pointer-generator networks [EB/OL]. (2017-04-25) . http://arxiv. org/abs/1704. 04368.

[16] Li Piji, Lam Wai, Bing Lidong. Deep recurrent generative decoder for abstractive text summarization [C]// Proc of Conference on Empirical Methods in Natural Language Processing. 2017: 2091–2100.